ISSN No. 0976-5697

Volume 4, No. 8, May-June 2013

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

Meiosis Based Cryptographic Hash Function Generation with the Influence of 3d Navier Stokes Equation on Chromasomal Chaotic Movement in Nucleoplasm.

Er. Bijoy Boban
School of Computer Science- K2, Dept. of Science and Technology
Lovely Professional University Phagwara, India
bijoy.boban7@gmail.com

Abstract: In this paper, we introduce a new system for cryptographic hash function generation with the help of molecular biology and to give light to a sector in computer science called Digital Cell Emulation Technique which can further be used for other purposes than the one which we have include in this paper; here we render the concept of meiotic cell division process, which is the chromosomal cross over undergone during fertilization which is the real reason for dissimilarity of an individual, genetically to another. We consider the latest finding on chaotic advection in micro-scale flow geometries which can be harnessed to greatly enhance the rate of thermally activated biochemical reactions like the polymerase chain reaction, well known as PCR. We use the above concept to emulate PCR during binary meiosis padding in the algorithm and we consider the state of the base pairs in the nucleoplasm and the motion that takes place in the fluid which we implement for crossover emulation by using 3D Navier Stokes equation with maximal convection speed of 0.055 micrometre/s during metaphase of meiosis. Even for slow-diffusing species such as ¸lambda-DNA the above process can be considered and so we can generalise it as the emulation of meiosis. Initial encryption is carried out which imitates the cross over and then the hash code generation that imitates the condensed chromosome formation. This algorithm ensures all the requirement standards for a good hash code algorithm and the pseudo code generator.

Keywords: Meiosis based cryptography, Nucleotide, Base pairs, Crossover, RC4, SHA512, Chaotic Motion, Rayleigh number, Navier-Strokes Equation, Continuum Mechanics, Rheology.

I. INTRODUCTION

Meiosis based Cryptographic hash function generator is a research topic which is not as explored as an individual event to be emulated although we have the genetic algorithm. The basis of intelligent life originates within the cell itself. The reactions of organic molecules resident in the cells give rise to highly efficient, intricate, self-correcting and self-directing processes such as replication, transcription and translation [1]. These processes, along with mutations form the basis of evolution, there by forming the foundation for the building of an intelligent life. These concepts form the basis of technological advancements for emulating computational intelligence in many engineering applications [2]. The prime reason for the dissimilarity of all the individuals who exist in this planet with different parental origin is chiasma formation during meiosis process of fertilization where the nucleotide strands from the mother and father undergoes cross over for the generation of 4 diploid cells.

If we can undergo the same principles of meiosis in data with a virtual construction of A-DNA from the data bits following the conventions of Cryptography and rendering of state at which the meiosis takes place, which include the centroidal based movement of the chromosome in the nucleoplasm which we can emulate with the help of 3D Chaotic flow states for Accelerated DNA replication in MICRO-SCALE convective PCR which uses 3D Navier-Stokes Equation in R^3 Euclidian space [3], then we can generate a highly secure Hash function that satisfies all the constrains of hash code and pseudo code. Security is a complex property and difficult to design or optimize. The existence of so many methods of attack makes the protection of an information system very complicated. The secrecy of transmitted information using encipherment and also with authentication of information verifying the identity of people, preventing the stealing of information and controlling access to both data and software have become such vital issues today, that data security is highly essential. In cryptography, the message is transformed so as to be unintelligible even though its existence is apparent [4].

Prior encryption with the help of chiasma formation is carried out before the hash code generation, the encryption consist of 'n' rounds which will be selected so that it support the non-retrieval or close encounter of the cipher code back to the plain code when implementing the encryption in 256 bit ASCII (we are starting the test with 128 rounds). Padding is done at every node with each node having a size of 1024 bits with 896 bit of data and 128 bit of padding, the first 8 bits of all the prime numbers from 3 to 1000 is used for the padding. Last 128 bit shows the length of the original message. The padding will be done on the basis of the emulation of the conditions for PCR during meiosis considering the onset of flow and transition to convective turbulence as determined by the dimensionless Rayleigh number ($Ra = [g\beta(T2–T1)h3]/\nu\alpha$; which explains the chaotic advection in micro-scale flow geometries which can be harnessed to greatly enhance the rate of thermally activated PCR [2]. For the encryption technique we use SHA512 like data handling with the ASCII intake similar to elliptical curve encryption which can be called an elliptical- modified version to the RC4 swapping. One of the main issues involved in cryptography is the existence of a global key [5]. A global key

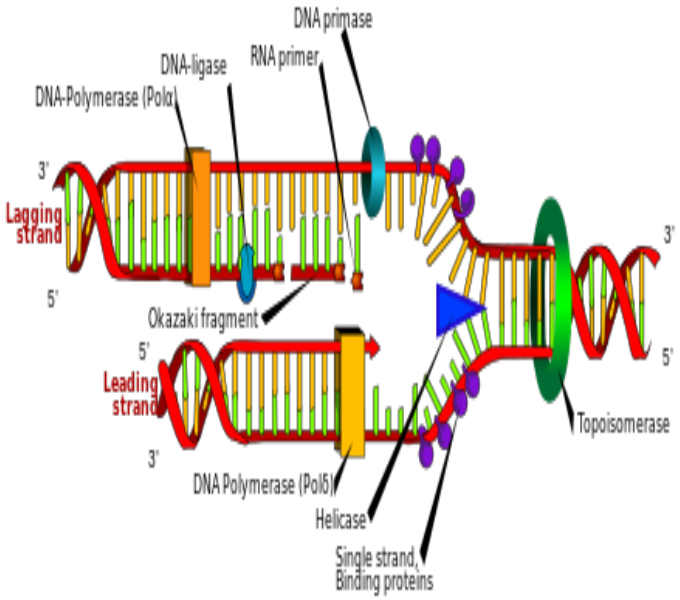
of appropriate length is chosen for the encryption process from which, at each stage, a sub-key is derived.

We use the Navier- Strokes Equation to emulate the cross over encryption, in Mathematics the Navier–Stokes equations are a system of nonlinear partial differential equations for abstract vector fields of any size, in physics and engineering, they are a system of equations that models the motion of liquids or non-rarefied gases using continuum mechanics. The equations are a statement of Newton's second law, with the forces modelled according to those in a viscous Newtonian fluid as the sum of contributions by pressure, viscous Stress and an external body force. Rheology is the study of the flow of matter, primarily in the liquid state, but also as 'soft solids' or solids under conditions in which they respond with plastic flow rather than deforming elastically in response to an applied force [3]. It applies to substances which have a complex molecular structure, such as mud, sludge, suspensions, polymers and other glass formers (e.g. silicates), as well as many foods and additives, bodily fluids (e.g. blood) and other biological materials. Newtonian fluids can be characterized by a single coefficient of viscosity for a specific temperature. Although this viscosity will change with temperature, it does not change with the flow rate or strain rate. Only a small group of fluids exhibit such constant viscosity, and they are known as Newtonian fluids. So we consider DNA as a Newtonian fluid. So it proves that we can use Navier- Strokes equation on DNA movement and we consider DNA as a Continuum Particle.

II. METHODS

There is not much research available on this subject and the only one available is an encryption technique which uses biological DNA sequence rendered from DNA pool to encrypt data in it and code it as image patterns in bitmap format [2]. But in out method we consider data in binary as a virtual DNA strand with a maximum length of 2^{128} bit size for plain text and for key it is 1024 bit.

Each strand of the DNA will be subdivided into pairs of 512 bits each and the base pairs will be represented with 64 bit of data from it. Chiasma formation and inter base pair transposition with variable acquisition strategy during the 128 rounds and the permutation of Key at the initial state to avoid the XOR nullification of the padded bits when doing XOR with the padded bits in the key and the plain which are the same containing the 1st 8 bits of the prime numbers from 3 to 1000 like the Fig 1.



DNA Replication process.

The padding will be undergone on the basis of the chaotic movement of the DNA and Rayleigh number ($Ra = [g\ \beta\ (T2 - T1)\ h3] / v\ \alpha$, i.e. at extension conditions (72 °C), emulating a yielding and an exponential increase in the number of copies of the target DNA sequence which is the padding in the current scenario [3] as in Fig 2.

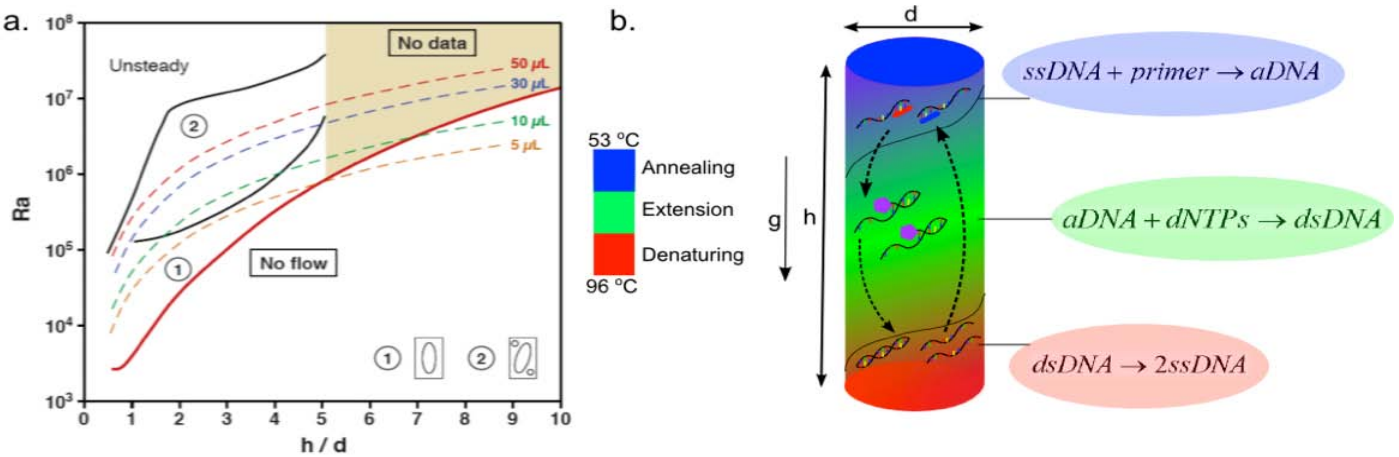


Figure: 2 Convectively actuated PCR in cylindrical reactor geometries. (a) Microscale thermal convection generates a multiplicity of flow regimes depending on Ra and h/d. (b) Denaturing, annealing, and extension reactions occur as fluid elements are continuously shuttled between different temperature zones in the cylindrical reactor [3].

We do a stepwise construction of virtual DNA strands and then do the process of meiosis as in Fig 3.

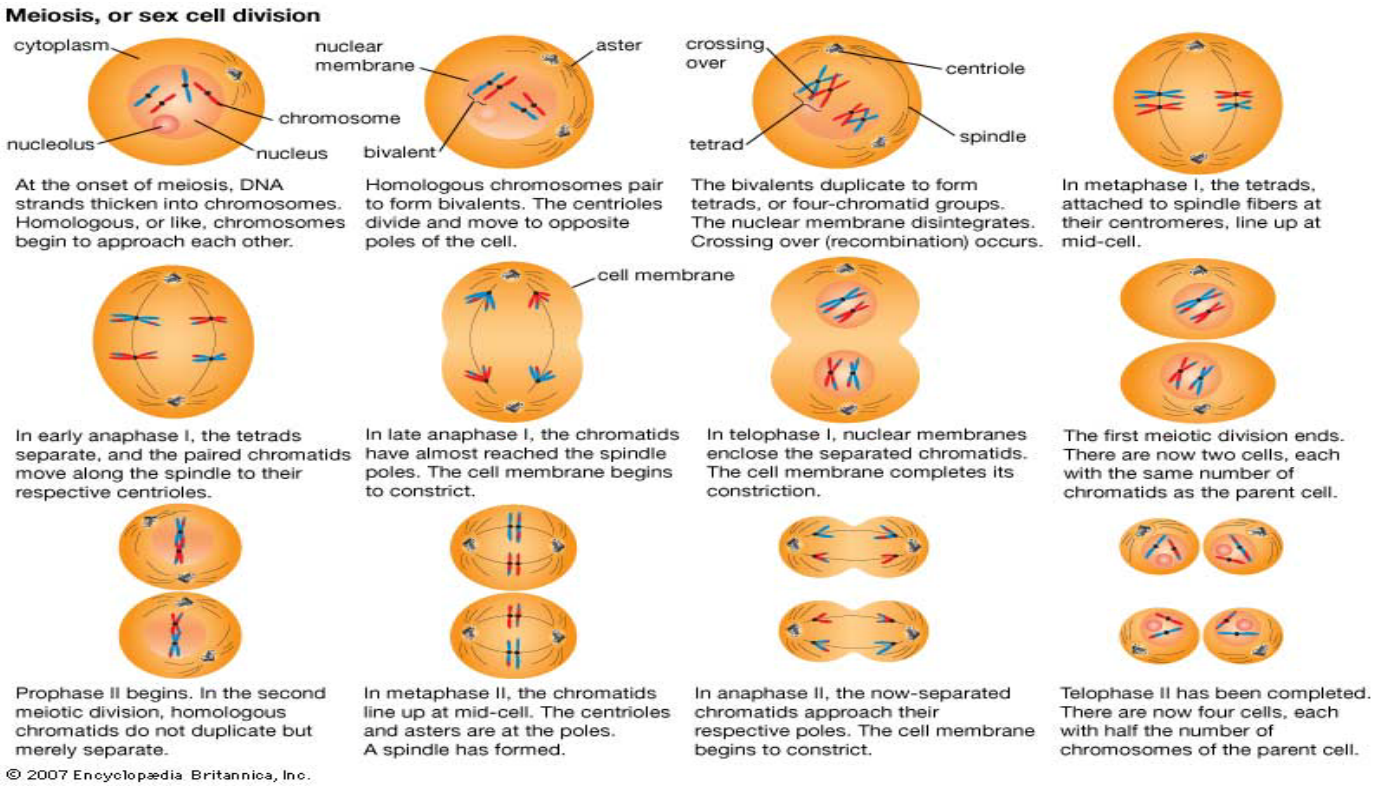


Figure: 3 Meiosis.

Cross over is the main process that makes the entire sequence permuted as in Fig 4.

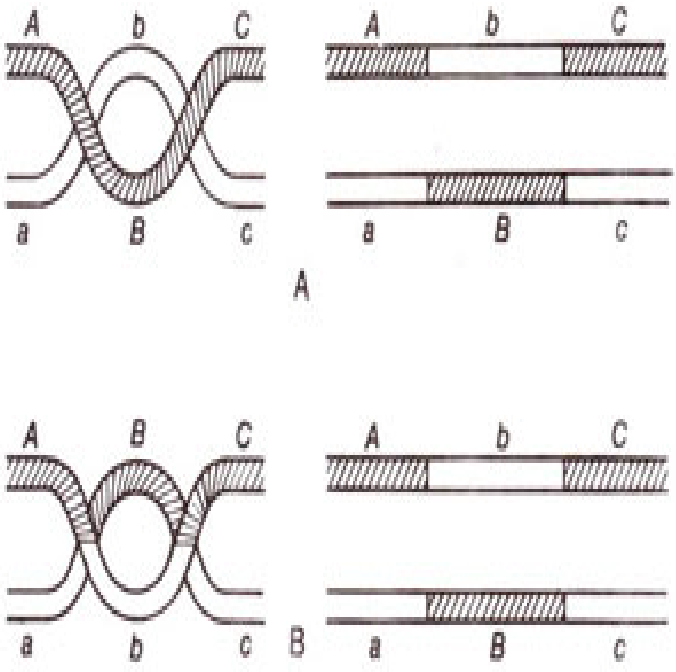


Figure: 4 Cross over.

The sample code of chiasma swap is as in Fig 5.

```
for(i=512;i<(y-512);i+=1024)
{
    for(j=i,k=i+1023;j<(i+512),k>((i+1023)-512);j+=128,k-=128)
    {
        for(l=j,m=k;l<(j+64),m>(k-64);l++,m--)
        {
            temp=node[l];
            node[l]=node[m];
            node[m]=temp;
        }
    }
}
```

Figure 5 Chiasma formation code.

We avoid the 1st and last 512 bits of the data bit as we do the chiasma of those bits with the key bits and it modifies the key as shown in Fig 6.

```
for(i=0,j=512;i<512,j<1024;i+=128,j+=128)
{
    for(k=i,l=j;k<(i+64),l<(j+64);k++,l++)
    {
        temp=node[k];
        node[k]=knode[l];
        knode[l]=temp;
    }
}
//left end done
//key modified
for(i=(y-1),j=0;i>((y-1)-512),j<512;i-=128,j+=128)
{
    for(k=i,l=j;k>(i-64),l<(j+64);k--,l++)
    {
        temp=node[k];
        node[k]=knode[l];
        knode[l]=temp;
    }
}
//right end done
```

Figure: 6 Chiasma modifies the key.

After the sequence of encryption we move on to the hash code, there we use the Navier- stokes equation; Here we assume, Let v(x, t) be a 3-dimensional vector, the velocity of the fluid, and let p(x, t) be the pressure of the fluid which we take as random prime numbers. The Navier–Stokes equations are supported by equation (1):

$$\delta v / \delta t + (v. \nabla)v = - \nabla p + \upsilon \Delta v + f(x, t) \quad (1)$$

Where v>0 is the kinematic viscosity, f(x, t) the external force, ∇ is the gradient operator and Δ is the Laplacian operator, which is also denoted by $\nabla.\nabla$. Note that this is a vector equation, i.e. it has three scalar equations. Then we write down the coordinates of the velocity and the external force as in equation (2) and equation (3).

$$V(x, t) = (\upsilon_1(x, t), \upsilon_2(x, t), v_3(x, t)) \quad (2)$$

$$f(x ,t) = (f_1(x, t),f_2(x, t),f_3(x, t)) \quad (3)$$

Then for each i=1,2,3 there is the corresponding scalar Navier–Stokes equation as shown is equation (4) is:

$$\frac{\partial v_i}{\partial t} + \sum_{j=1}^{3} v_j \frac{\partial v_i}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \nu \sum_{j=1}^{3} \frac{\partial^2 v_i}{\partial x_j^2} + f_i(\boldsymbol{x}, t). \quad (4)$$

This Equation is then used to initiate the hash code generation, after that we will make an isomer for the 1st 512 DNA bit sequence so as to make is satisfy the constrains of good pseudo code, then we permute the code with the prime or the key according to the polynomial equation based condition like a polymerase, and then the hash code is generated after performing 54 rounds of the above steps as we use 54 prime numbers. Actually the polymerase round alter the equality formed during isomer round in the totality of 0's and 1's in the binary sequence as shown in Fig 7.

```
y=1024;
for(i=0,j=(y/3);i<y;i++,j=(j+((i^7)%1024))%1024)
{
    if(node[i]!=knode[j])
    {
        temp=node[(i*j)%y];
        node[(i*j)%y]=node[i];
        node[i]=temp;
    }
    if(node[(i+j)%y]!=prime[i%128][j%8])
    {
        temp=prime[i%128][j%8];
        node[(i+j)%y]=node[i];
        node[i]=temp;
    }
    if(i==y-1)
        break;
}//polymerase made
```

Figure: 7 Polymerase Formation code.

After this step, the 1024 bit hash code will be generated. The number of 0's and 1's in the hash code will be approximately equal as we do the polymerase code which satisfies the requirement of a good pseudo code generator. The algorithm is simple in concept and can be easily used, but the susceptibility for cracking is much less.

III. CONCLUSION

As it is a research topic and as there are only few papers published on this topic there is only little information available on the topic and as the 3D Navier- Stokes equation is not cracked till date it can support the paper in securing the security constrain and the rendering of meiosis for encryption and hash code generation also provide a new stream of cryptographic methodology inside DNA based cryptography and we conclude that with this method we can generate a highly secure 1024 bit hash code that satisfies all the constrains for a good hash code and a pseudo code. The algorithm is complex but is optimised for fast execution.

IV. ACKNOWLEDGMENT

The topic was inter disciplinary, which included not just cryptography but also the basics of fertilization and mutation, fluid mechanics, continuum mechanics etc, I thank Asst. Prof Ankur Sodhi and Asst. Prof Gour Sundar Mitra Takur for help they provided in sorting out the problem and for the other technical support. I deliver my thanks to Asst. Prof Raji Ravi of Saintgits collage of engineering, Pathamuttom, Kottayam, Kerala, India for the inter disciplinary information's he provided. I extend my thanks to the department of Computer science in Lovely Professional University.

V. REFERENCES

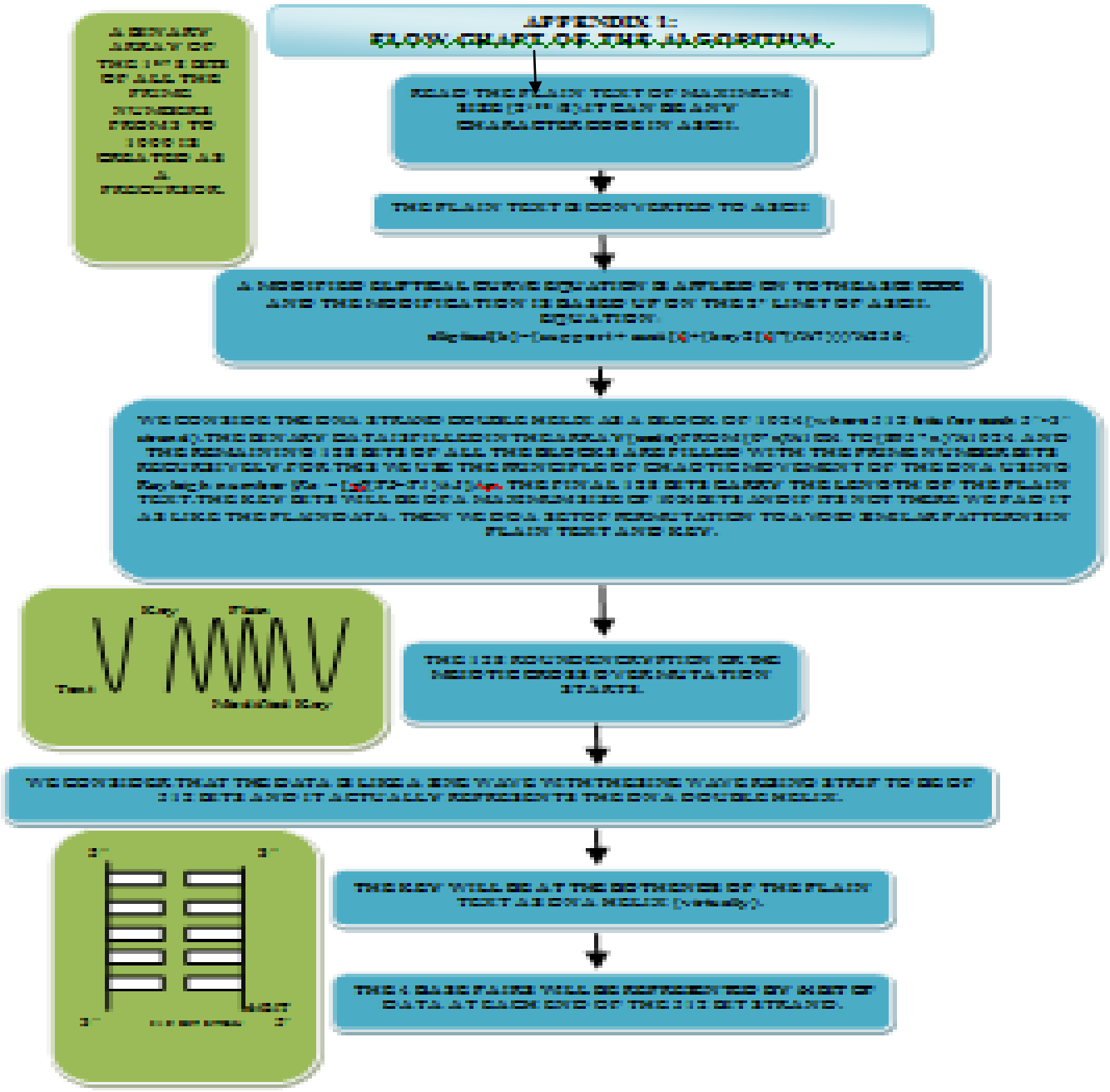
[1] Herskowitz,H. I., Basic Principles of Molecular Genetics, Thomas Nelson, England, 1968.

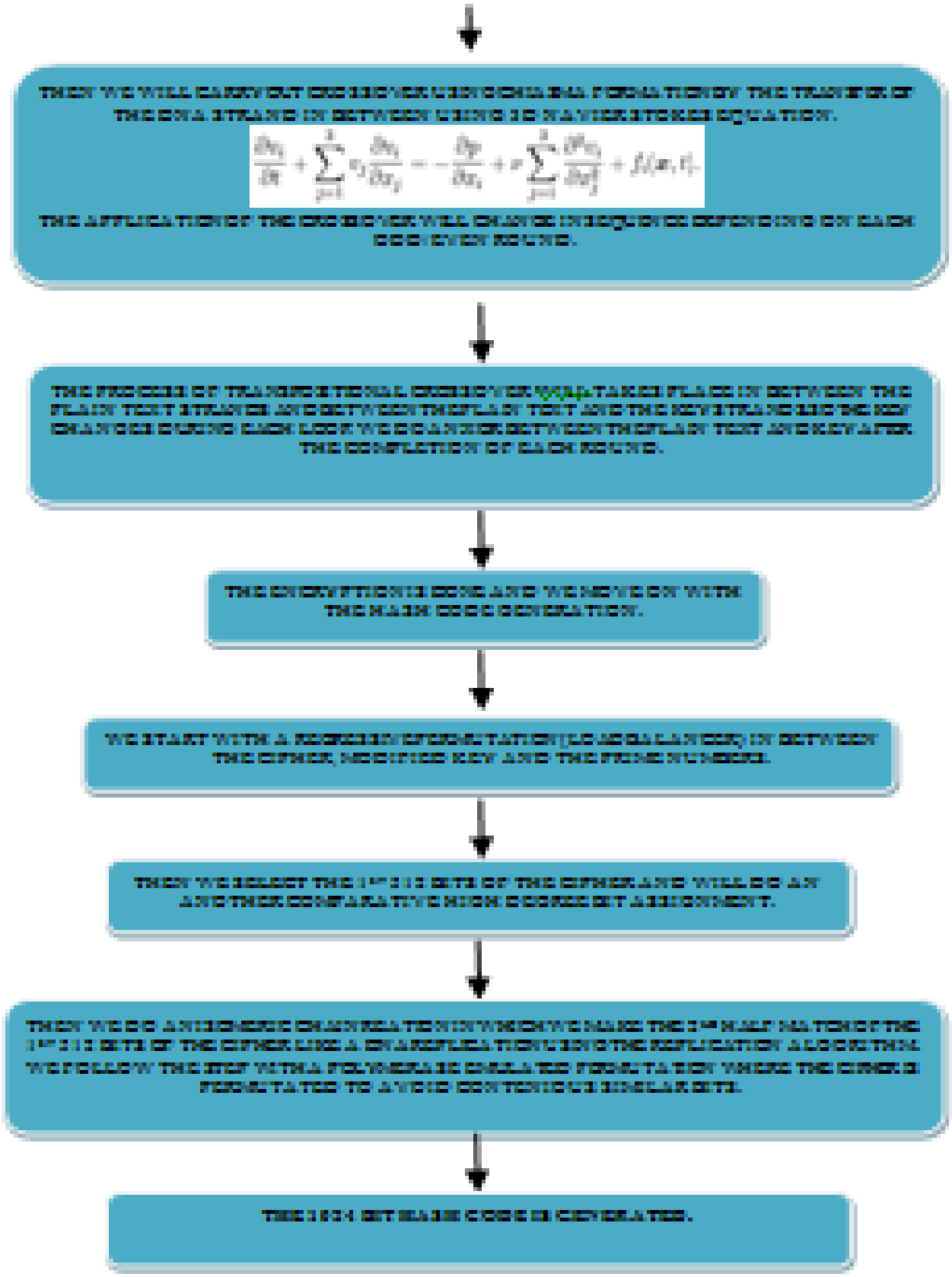
[2] M. Prashant, R. Siddharth and Rajeev Kumar , Formulation of an Encryption Algorithm on the Basis of Molecular Genetics and Image Patterns, Department of Computer Science & Information Systems, Birla Institute of Technology & Science, Pilani – 333 031, India.

[3] Aashish Priye, Radha Muddu, Yassin A. Hassan, and Victor M, Selecting 3D Chaotic flow states for Accelerated DNA replication in MICRO-SCALE convective PCR, Ugaz, Department of Chemical Engineering, Texas A&M University, College Station, TX 77843, USA.

[4] Thomas Y. Hou, Zhen Lei, On the Partial Regularity of a 3D Model of the Navier-Stokes Equations, 1. Applied and Comput. Math, Caltech, Pasadena, CA 91125, USA. 2. School of Mathematical Sciences, Fudan University, Shanghai 200433, P.R. China.

[5] H. Feistel, “Cryptography and computer privacy”, Scientific American, vol. 228, no. 5, p 15, May-1973.





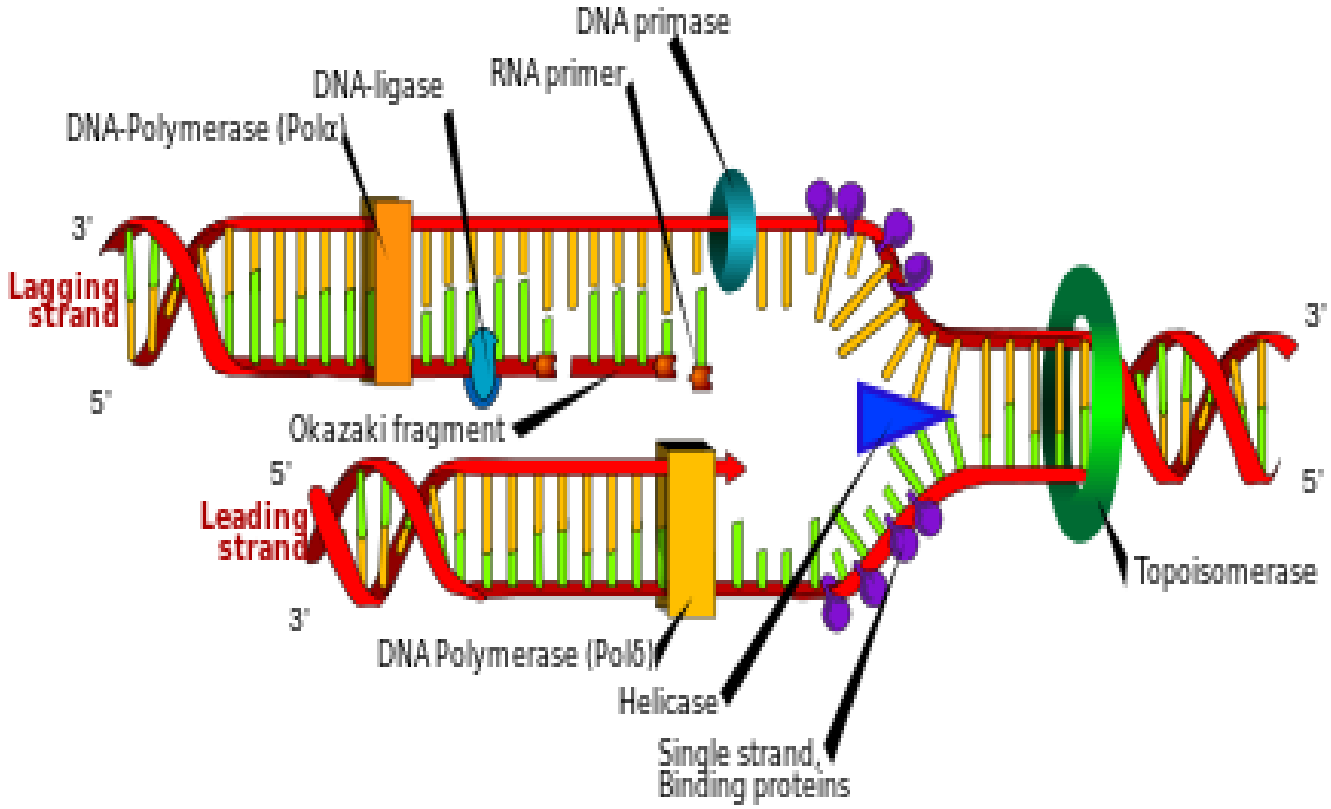


Figure 1: DNA Replication Process.

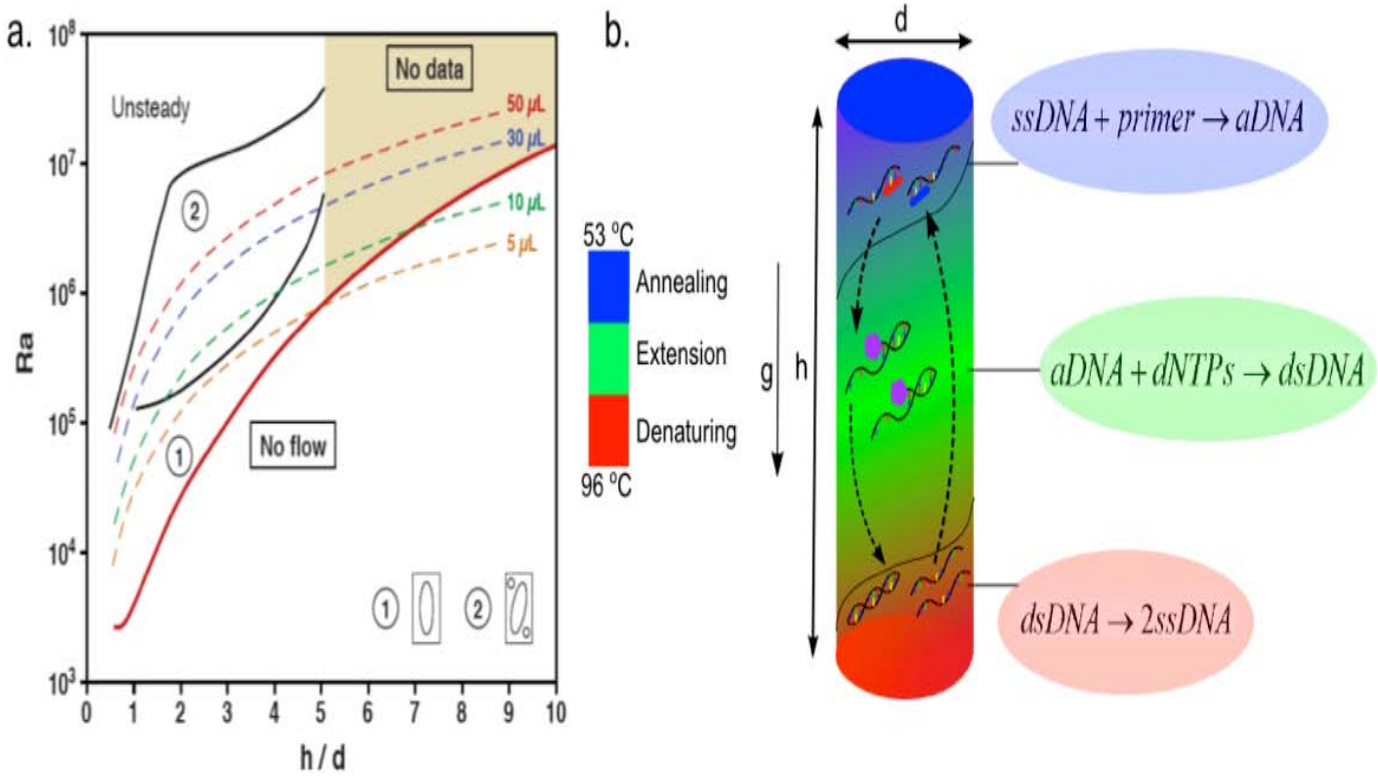


Figure 2: Convectively actuated PCR in cylindrical reactor geometries. (a) Micro scale thermal convection generates a multiplicity of flow regimes depending on Ra and h/d. (b) Denaturing, annealing, and extension reactions occur as fluid elements are continuously shuttled between different temperature zones in the cylindrical reactor.

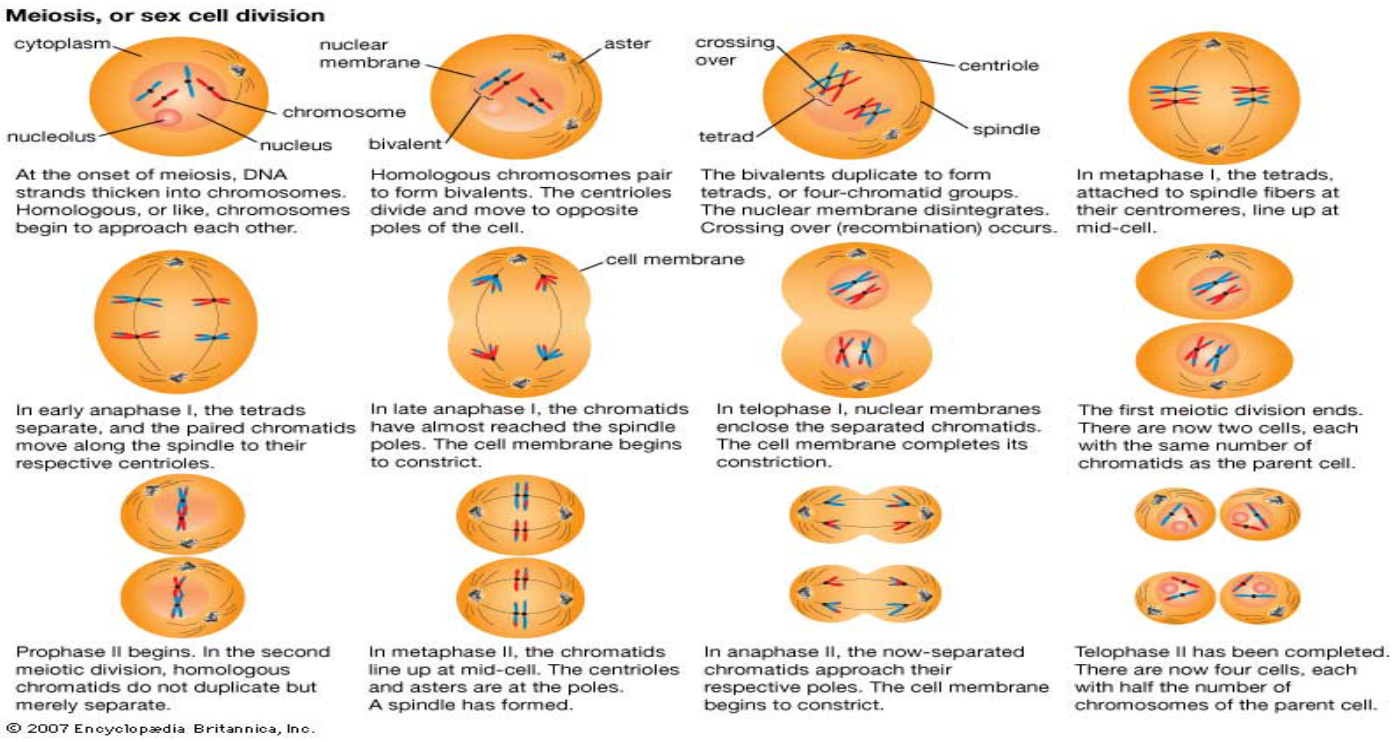


Figure 3: Meiosis

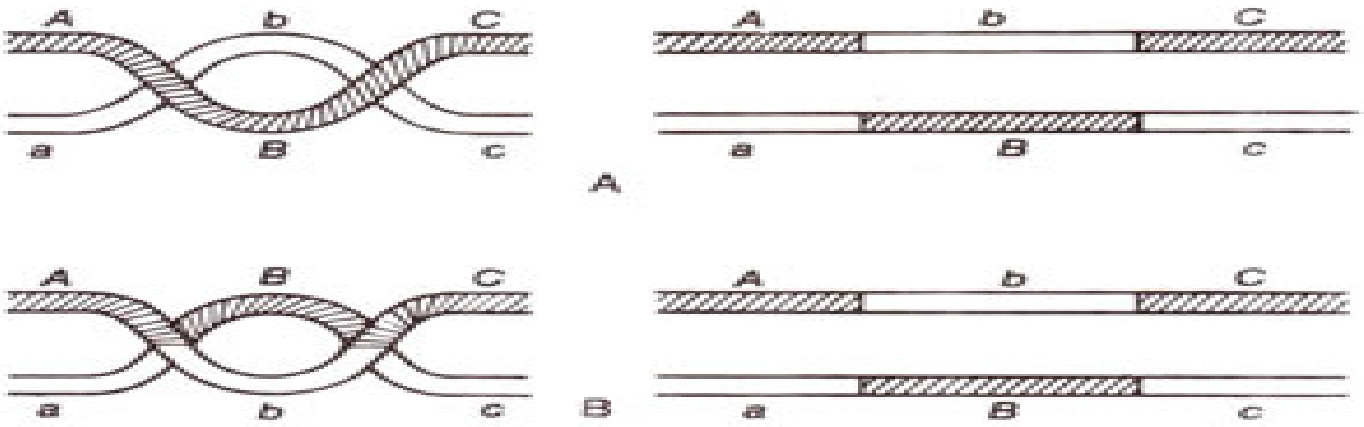


Figure 4: Cross over

```
for(i=512;i<(y-512);i+=1024)
{
    for(j=i,k=i+1023;j<(i+512),k>((i+1023)-512);j+=128,k-=128)
    {
        for(l=j,m=k;l<(j+64),m>(k-64);l++,m--)
        {
            temp=node[l];
            node[l]=node[m];
            node[m]=temp;
        }
    }
}
```

Figure 5: Chiasma formation code.

```
for(i=0,j=512;i<512,j<1024;i+=128,j+=128)
{
    for(k=i,l=j;k<(i+64),l<(j+64);k++,l++)
    {
        temp=node[k];
        node[k]=knode[l];
        knode[l]=temp;
    }
}
//left end done
//key modified
for(i=(y-1),j=0;i>((y-1)-512),j<512;i-=128,j+=128)
{
    for(k=i,l=j;k>(i-64),l<(j+64);k--,l++)
    {
        temp=node[k];
        node[k]=knode[l];
        knode[l]=temp;
    }
}
//right end done
```

Figure 6: Chiasma modifies the key.

```
y=1024;
for(i=0,j=(y/3);i<y;i++,j=(j+((i^7)%1024))%1024)
{
    if(node[i]!=knode[j])
    {
        temp=node[(i*j)%y];
        node[(i*j)%y]=node[i];
        node[i]=temp;
    }
    if(node[(i+j)%y]!=prime[i%128][j%8])
    {
        temp=prime[i%128][j%8];
        node[(i+j)%y]=node[i];
        node[i]=temp;
    }
    if(i==y-1)
        break;
}//polymerase made
```

Figure 7: Polymerase formation code.